

1. Method of structurally converting a binary sequence into an encrypted final image  $G$ , the structural conversion comprising the steps of:

forming an image  $M$  of the binary sequence as a concatenation of a tag data element  $T$  and structural data element  $S$ , tag data element  $T$  comprising information necessary to reverse a conversion process, structural data element  $S$  comprising a sequence of logical scales of position coding;

selecting a number of conversion function iterations  $P$  to be performed;

iteratively executing  $P$  times a conversion function comprised of the following steps:

- selecting a transformation algorithm  $A$  from a predefined set of transformation algorithms  $L$ ;
- selecting an alphabet of transformation  $AV$  based upon the structural data element  $S$ ;
- applying algorithm  $A$  and alphabet  $AV$  to structural data element  $S$  to form a plurality of logical scales of position coding;
- forming a transformed structural data element  $S'$  comprised of a sequence of the logical scales of position coding;
- selecting an external key  $K^x$ ;
- forming tag data element  $T$ ;
- coding the tag data element  $T$  with external key  $K^x$  to obtain coded tag data element  $T''$ ;
- repeating the steps of the conversion function upon a converted image  $M'$  comprised of a concatenation of the coded tag data element  $T''$  and the transformed structural data element  $S'$ ;

and forming the encrypted final image  $G$  as a concatenation of the coded tag data element  $T''$  and the transformed structural data element  $S'$  created upon the  $P^{th}$

1 iteration of the conversion function.

2

3 2. Method of structurally converting a binary sequence into an encrypted final image

4  $G$ , said structural conversion comprising the steps of:

5

6 forming an image  $M$  of the binary sequence as a concatenation of a tag data

7 element  $T$  and structural data element  $S$ , tag data element  $T$  comprising

8 information necessary to reverse the conversion process, structural data element  $S$

9 comprising a sequence of logical scales of position coding;

10

11 selecting a number of conversion function iterations  $P$  to be performed;

12

13 iteratively executing  $P$  times a conversion function comprised of the following

14 steps:

15 selecting a transformation algorithm  $A$  from a predefined set of

16 transformation algorithms  $L$ ;

17 selecting an alphabet of transformation  $AV$  based upon the structural data

18 element  $S$ ;

19 applying algorithm  $A$  and alphabet  $AV$  to structural data element  $S$  to form

20 a plurality of logical scales of position coding;

21 forming a transformed structural data element  $S'$  comprised of a sequence

22 of the logical scales of position coding;

23 stochastically selecting a bit length parameter and a shift parameter which

24 define an internal identifier  $K$  within transformed structural data

25 element  $S'$ ;

26 forming tag data element  $T$ ;

27 coding a portion of the tag data element  $T$  with internal identifier  $K$  to

28 obtain a partially coded tag data element  $T'$ ;

29 selecting an external key  $K^x$ ;

30 coding the partially coded tag data element  $T'$  with external key  $K^x$  to

1                   obtain coded tag data element  $T''$ ;  
2           determining whether to extract internal identifier  $K$  from  
3           transformed structural data element  $S'$ , and if determined  
4           necessary, extracting the internal identifier  $K$  from transformed  
5           structural data element  $S'$  to obtain structural data element  $S''$  and  
6           storing internal identifier  $K$  in a file of internal identifiers  $FID$ ;  
7           repeating the steps of the conversion function upon a converted image  $M'$   
8           comprised of a concatenation of the coded tag data element  $T''$  and  
9           either transformed structural data element  $S'$  if internal identifier  $K$   
10          was not extracted, or structural data element  $S''$  if internal identifier  
11           $K$  was extracted;

12  
13           and forming the encrypted final image  $G$  as a concatenation of the coded tag  
14           data element  $T''$  and either transformed structural data element  $S'$  if internal  
15           identifier  $K$  was not extracted, or structural data element  $S''$  if internal  
16           identifier  $K$  was extracted.

17  
18    3.    Method of structurally converting a binary sequence into an encrypted final image  
19     $G$ , said structural conversion comprising the steps of:

20  
21           forming an image  $M$  of the binary sequence as a concatenation of a tag data  
22           element  $T$  and structural data element  $S$ , tag data element  $T$  comprising  
23           information necessary to reverse the conversion process, structural data element  $S$   
24           comprising a sequence of logical scales of position coding;

25  
26           selecting a number of conversion function iterations  $P$  to be performed;

27  
28           iteratively executing  $P$  times a conversion function comprised of the following  
29           steps:

30           selecting a transformation algorithm  $A$  from a predefined set of

1 transformation algorithms  $L$ ;  
 2 selecting an alphabet of transformation  $AV$  based upon the structural data  
 3 element  $S$ ;  
 4 applying algorithm  $A$  and alphabet  $AV$  to structural data element  $S$  to form  
 5 a plurality of logical scales of position coding;  
 6 forming a transformed structural data element  $S'$  comprised of a sequence  
 7 of the logical scales of position coding;  
 8 stochastically selecting a bit length parameter and a shift parameter which  
 9 define an internal identifier  $K$  within transformed structural data  
 10 element  $S'$ ;  
 11 scrambling internal identifier  $K$  with a scrambling function to obtain a  
 12 scrambled internal identifier  $K'$ ;  
 13 forming tag data element  $T$ ;  
 14 coding a portion of the tag data element  $T$  with scrambled internal  
 15 identifier  $K'$  to obtain a partially coded tag data element  $T'$ ;  
 16 selecting an external key  $K^x$ ;  
 17 coding the partially coded tag data element  $T'$  with external key  $K^x$  to  
 18 obtain coded tag data element  $T''$ ;  
 19 determining whether to extract internal identifier  $K$  from  
 20 transformed structural data element  $S'$ , and if determined  
 21 necessary, extracting the internal identifier  $K$  from transformed  
 22 structural data element  $S'$  to obtain structural data element  $S''$  and  
 23 storing scrambled internal identifier  $K'$  in a file of internal  
 24 identifiers  $FID$ ;  
 25 repeating the steps of the conversion function upon a converted image  $M'$   
 26 comprised of a concatenation of the coded tag data element  $T''$  and  
 27 either transformed structural data element  $S'$  if internal identifier  $K$   
 28 was not extracted, or structural data element  $S''$  if internal  
 29 identifier  $K$  was extracted;  
 30

- 1 and forming the encrypted final image  $G$  as a concatenation of the coded tag data  
2 element  $T''$  and either transformed structural data element  $S'$  if internal identifier  
3  $K$  was not extracted, or structural data element  $S''$  if internal identifier  $K$  was  
4 extracted.
- 5
- 6 4. The method of claim 2, further comprising the steps of:
- 7
- 8 structurally converting the file of internal identifiers  $FID$  to obtain a converted file  
9 of internal identifiers  $FID'$ , wherein a tag data element formed during the  
10 structural conversion of the file of internal identifiers  $FID$  is coded with an  
11 external key selected stochastically from a multitude of external keys in an  
12 external key file  $K_{EXT}$ ; and
- 13
- 14 optionally transmitting the encrypted final image  $G$  and structurally converted file  
15 of internal identifiers  $FID'$  to a subscriber or receiver.
- 16
- 17 5. The method of claim 1, wherein the external key  $K^x$  is selected from a multitude  
18 of external keys in an external key file  $K_{EXT}$ .
- 19
- 20 6. The method of claim 1, wherein the selection of external key  $K^x$  is a stochastic  
21 selection.
- 22
- 23 7. The method of claim 1, wherein a same external key  $K^x$  is selected for use in all  
24 iterations.
- 25
- 26 8. The method of claim 1, wherein a different external key  $K^x$  is selected upon each  
27 iteration.
- 28
- 29 9. The method of claim 1, wherein the external key  $K^x$  is entered by a user during the  
30 conversion and reverse conversion process.

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30

10. The method of claim 5, further comprising the steps of:
- structurally converting the external key file  $K_{EXT}$  to obtain a structurally converted external key file; and
- transmitting to a subscriber the structurally converted external key file and an initial key  $K_{INIT}$  required to reverse the structural conversion of the structurally converted external key file to obtain the external key file  $K_{EXT}$ .
11. The method of claim 1, wherein the selection of transformation algorithm  $A$  may be a stochastic selection.
12. The method of claim 1, wherein the selection of transformation algorithm  $A$  may depend upon adherence to a mathematical criterion.
13. The method of claim 1, wherein the selection of transformation algorithm  $A$  may depend upon adherence to a logical criterion.
14. The method of claim 1, wherein the selection of transformation algorithm  $A$  may depend upon adherence to a file size criteria for encrypted final image  $G$ .
15. The method of claim 1, wherein the predefined set of transformation algorithms  $L$  may be supplemented.
16. The method of claim 1, wherein the selection of a number of conversion steps  $P$  may be a stochastic selection.
17. The method of claim 1, wherein the selection of a number of conversion steps  $P$  may depend upon adherence to a mathematical criterion.

- 1 18. The method of claim 1, wherein the selection of a number of conversion steps  $P$   
2 may depend upon adherence to a logical criterion.  
3
- 4 19. The method of claim 1, wherein the selection of a number of conversion steps  $P$   
5 may depend upon adherence to a file size criteria for encrypted final image  $G$ .  
6
- 7 20. The method of claim 1, wherein the alphabet of transformation  $AV$  is comprised  
8 of letters or quants, each letter or quant comprising a segment of structural data  
9 element  $S$ .  
10
- 11 21. The method of claim 2, further comprising the step of determining upon which  
12 iterations, if any, internal identifiers are to be extracted.  
13
- 14 22. The method of claim 3, further comprising the step of determining upon which  
15 iterations, if any, internal identifiers are to be extracted.  
16
- 17 23. The method of claim 20, wherein a number of bits in each letter or quant is  
18 stochastically selected.  
19
- 20 24. The method of claim 20, wherein a number of bits in each letter or quant may  
21 depend upon adherence to a mathematical criterion.  
22
- 23 25. The method of claim 20, wherein a number of bits in each letter or quant may  
24 depend upon adherence to a logical criterion.  
25
- 26 26. The method of claim 20, wherein a number of bits in each letter or quant may  
27 depend upon adherence to a file size criteria for encrypted final image  $G$ .  
28

1 27. The method of claim 1, wherein the information necessary to reverse the  
2 conversion process stored in tag data element  $T$  may comprise one or more of the  
3 following:

4  
5 an indicator of whether a current iterative step is the  $P^{th}$  iteration;

6  
7 an indicator of whether the selected external key  $K^x$  is to be used for all  $P$   
8 iterations;

9 an indicator of the selected external key  $K^x$ ;

10  
11 an indicator of the selected transformation algorithm  $A$ ;

12  
13 a length of a first logical scale of position coding;

14 an indicator of user information;

15  
16 the alphabet of transformation  $AV$ ; and

17  
18 other transformation algorithm  $A$  parameters.  
19

20 28. The method of claim 2, wherein the information necessary to reverse the  
21 conversion process stored in tag data element  $T$  may comprise one or more of the  
22 following:

23  
24 an indicator of whether a current iterative step is the  $P^{th}$  iteration;

25  
26 an indicator of whether the selected external key  $K^x$  is to be used for all  $P$   
27 iterations;

28  
29 an indicator of the selected external key  $K^x$ ;  
30



1 an indicator of the selected transformation algorithm  $A$ ;  
2 an indicator of user information;  
3  
4 the alphabet of transformation  $AV$ ;  
5  
6 a length of a first logical scale of position coding;  
7  
8 other transformation algorithm  $A$  parameters;  
9  
10 the bit internal identifier  $K$  length and shift parameters; and  
11 an indicator of internal identifier  $K$  extraction.  
12

13 29. The method of claim 3, wherein the information necessary to reverse the  
14 conversion process stored in tag data element  $T$  may comprise one or more of the  
15 following:  
16

17 an indicator of whether a current iterative step is the  $P^{th}$  iteration;  
18

19 an indicator of whether the selected external key  $K^x$  is to be used for all  $P$   
20 iterations;  
21

22 an indicator of the selected external key  $K^x$ ;  
23

24 an indicator of the selected transformation algorithm  $A$ ;  
25 an indicator of user information;  
26

27 the alphabet of transformation  $AV$ ;  
28

29 a length of a first logical scale of position coding;  
30

- 1                   other transformation algorithm  $A$  parameters;  
2  
3                   an indicator of the scrambling function selected;  
4  
5                   the bit internal identifier  $K$  length and shift parameters; and  
6  
7                   an indicator of internal identifier  $K$  extraction.  
8
- 9    30.    The method of claim 3, wherein the scrambling function is selected from a  
10           scrambling matrix comprised of a predefined set of scrambling functions.  
11
- 12   31.    The method of claim 30, wherein the predefined set of scrambling functions is  
13           changed periodically.  
14
- 15   32.    The method of claim 1, wherein the conversion function further comprises the  
16           step of:  
17           determining whether to insert user information into structural data element  $S$ , and  
18           inserting user information into structural data element  $S$  if determined necessary,  
19           thereby providing a means for user authentication and digital signing.  
20
- 21   33.    The method of claim 2, wherein the conversion function further comprises the  
22           step of:  
23           determining whether to insert user information into structural data element  $S$ , and  
24           inserting user information into structural data element  $S$  if determined necessary,  
25           thereby providing a means for user authentication and digital signing.  
26
- 27   34.    The method of claim 3, wherein the conversion function further comprises the  
28           step of:

1 determining whether to insert user information into structural data element  $S$ , and  
2 inserting user information into structural data element  $S$  if determined necessary,  
3 thereby providing a means for user authentication and digital signing.  
4

5 35. Computer executable process steps stored on a computer readable medium, the  
6 computer executable process steps for structurally converting a binary sequence  
7 into an encrypted final image  $G$ , the computer executable process steps  
8 comprising:  
9

10 forming an image  $M$  of the binary sequence as a concatenation of a tag data  
11 element  $T$  and structural data element  $S$ , tag data element  $T$  comprising  
12 information necessary to reverse a conversion process, structural data element  $S$   
13 comprising a sequence of logical scales of position coding;  
14

15 selecting a number of conversion steps  $P$  to be performed;  
16

17 iteratively executing  $P$  times a conversion function comprised of the following  
18 steps:  
19 selecting a transformation algorithm  $A$  from a predefined set of  
20 transformation algorithms  $L$ ;  
21 selecting an alphabet of transformation  $AV$  based upon the structural data  
22 element  $S$ ;  
23 applying algorithm  $A$  and alphabet  $AV$  to structural data element  $S$  to form  
24 a plurality of logical scales of position coding;  
25 forming a transformed structural data element  $S'$  comprised of a sequence  
26 of the logical scales of position coding;  
27 selecting an external key  $K^x$ ;  
28 forming tag data element  $T$ ;  
29 coding the tag data element  $T$  with external key  $K^x$  to obtain coded tag  
30 data element  $T''$ ;

1 repeating the steps of the conversion function upon a converted image  $M'$   
2 comprised of a concatenation of the coded tag data element  $T''$  and the  
3 transformed structural data element  $S'$ ;

4  
5 and forming the encrypted final image  $G$  as a concatenation of the coded tag data  
6 element  $T''$  and the transformed structural data element  $S'$  created upon the  $P^{th}$   
7 iteration of the conversion function.

8  
9 36. Computer executable process steps stored on a computer readable medium, the  
10 computer executable process steps for structurally converting a binary sequence  
11 into an encrypted final image  $G$ , the computer executable process steps  
12 comprising:

13  
14 forming an image  $M$  of the binary sequence as a concatenation of a tag data  
15 element  $T$  and structural data element  $S$ , tag data element  $T$  comprising  
16 information necessary to reverse the conversion process, structural data element  $S$   
17 comprising a sequence of logical scales of position coding;

18  
19 selecting a number of conversion steps  $P$  to be performed;

20  
21 iteratively executing  $P$  times a conversion function comprised of the following  
22 steps:

23 selecting a transformation algorithm  $A$  from a predefined set of  
24 transformation algorithms  $L$ ;

25 selecting an alphabet of transformation  $AV$  based upon the structural data  
26 element  $S$ ;

27 applying algorithm  $A$  and alphabet  $AV$  to structural data element  $S$  to form  
28 a plurality of logical scales of position coding;

29 forming a transformed structural data element  $S'$  comprised of a sequence  
30 of the logical scales of position coding;

1                   stochastically selecting a bit length parameter and a shift parameter which  
 2                   define an internal identifier  $K$  within transformed structural data  
 3                   element  $S'$ ;  
 4                   forming tag data element  $T$ ;  
 5                   coding a portion of the tag data element  $T$  with internal identifier  $K$  to  
 6                   obtain a partially coded tag data element  $T'$ ;  
 7                   selecting an external key  $K^x$ ;  
 8                   coding the partially coded tag data element  $T'$  with external key  $K^x$  to  
 9                   obtain coded tag data element  $T''$ ;  
 10                  stochastically determining whether to extract internal identifier  $K$  from  
 11                  transformed structural data element  $S'$ , and if determined  
 12                  necessary, extracting the internal identifier  $K$  from transformed  
 13                  structural data element  $S'$  to obtain structural data element  $S''$  and  
 14                  storing internal identifier  $K$  in a file of internal identifiers  $FID$ ;  
 15                  performing the steps of the conversion function upon a converted image  
 16                   $M'$  comprised of a concatenation of the coded tag data element  $T''$   
 17                  and either transformed structural data element  $S'$  if internal  
 18                  identifier  $K$  was not extracted, or structural data element  $S''$  if  
 19                  internal identifier  $K$  was extracted;  
 20  
 21                  and forming the encrypted final image  $G$  as a concatenation of the coded tag data  
 22                  element  $T''$  and either transformed structural data element  $S'$  if internal identifier  
 23                   $K$  was not extracted, or structural data element  $S''$  if internal identifier  $K$  was  
 24                  extracted.  
 25  
 26    37.    Computer executable process steps stored on a computer readable medium, the  
 27            computer executable process steps for structurally converting a binary sequence  
 28            into an encrypted final image  $G$ , the computer executable process steps  
 29            comprising:  
 30

1 forming an image  $M$  of the binary sequence as a concatenation of a tag data  
 2 element  $T$  and structural data element  $S$ , tag data element  $T$  comprising  
 3 information necessary to reverse the conversion process, structural data element  $S$   
 4 comprising a sequence of logical scales of position coding;  
 5  
 6 selecting a number of conversion steps  $P$  to be performed;  
 7  
 8 iteratively executing  $P$  times a conversion function comprised of the following  
 9 steps:  
 10       selecting a transformation algorithm  $A$  from a predefined set of  
 11               transformation algorithms  $L$ ;  
 12       selecting an alphabet of transformation  $AV$  based upon the structural data  
 13               element  $S$ ;  
 14       applying algorithm  $A$  and alphabet  $AV$  to structural data element  $S$  to form  
 15               a plurality of logical scales of position coding;  
 16       forming a transformed structural data element  $S'$  comprised of a sequence  
 17               of the logical scales of position coding;  
 18       stochastically selecting a bit length parameter and a shift parameter which  
 19               define an internal identifier  $K$  within transformed structural data  
 20               element  $S'$ ;  
 21       scrambling internal identifier  $K$  with a scrambling function to obtain a  
 22               scrambled internal identifier  $K'$ ;  
 23       forming tag data element  $T$ ;  
 24       coding a portion of the tag data element  $T$  with scrambled internal  
 25               identifier  $K'$  to obtain a partially coded tag data element  $T'$ ;  
 26       selecting an external key  $K^x$ ;  
 27       coding the partially coded tag data element  $T'$  with external key  $K^x$  to  
 28               obtain coded tag data element  $T''$ ;  
 29       stochastically determining whether to extract internal identifier  $K$  from  
 30               transformed structural data element  $S'$ , and if determined

1                   necessary, extracting the internal identifier  $K$  from transformed  
2                   structural data element  $S'$  to obtain structural data element  $S''$  and  
3                   storing scrambled internal identifier  $K'$  in a file of internal  
4                   identifiers  $FID$ ;  
5                   performing the steps of the conversion function upon a converted image  
6                    $M'$  comprised of a concatenation of the coded tag data element  $T''$   
7                   and either transformed structural data element  $S'$  if internal  
8                   identifier  $K$  was not extracted, or structural data element  $S''$  if  
9                   internal identifier  $K$  was extracted;

10  
11                   and forming the encrypted final image  $G$  as a concatenation of the coded tag data  
12                   element  $T''$  and either transformed structural data element  $S'$  if internal identifier  
13                    $K$  was not extracted, or structural data element  $S''$  if internal identifier  $K$  was  
14                   extracted.

15  
16    38.    The computer executable process steps stored on a computer readable medium of  
17           claim 35, wherein the external key  $K^x$  is selected from a multitude of external  
18           keys in an external key file  $K_{EXT}$ .

19  
20    39.    The computer executable process steps stored on a computer readable medium of  
21           claim 35, wherein the selection of the external key  $K^x$  is a stochastic selection.

22  
23    40.    The computer executable process steps stored on a computer readable medium of  
24           claim 35, wherein a same external key  $K^x$  is selected for use in all iterations.

25  
26    41.    The computer executable process steps stored on a computer readable medium of  
27           claim 35, wherein a different external key  $K^x$  is selected upon each iteration.

28

- 1 42. The computer executable process steps stored on a computer readable medium of  
2 claim 35, wherein the external key  $K^x$  is entered by a user during the conversion  
3 and reverse conversion process.  
4
- 5 43. The computer executable process steps stored on a computer readable medium of  
6 claim 35, wherein the selection of transformation algorithm  $A$  may be a stochastic  
7 selection.  
8
- 9 44. The computer executable process steps stored on a computer readable medium of  
10 claim 35, wherein the selection of transformation algorithm  $A$  may depend upon  
11 adherence to a mathematical criterion.  
12
- 13 45. The computer executable process steps stored on a computer readable medium of  
14 claim 35, wherein the selection of transformation algorithm  $A$  may depend upon  
15 adherence to a logical criterion.  
16
- 17 46. The computer executable process steps stored on a computer readable medium of  
18 claim 35, wherein the selection of transformation algorithm  $A$  may depend upon  
19 adherence to a file size criteria for encrypted final image  $G$ .  
20
- 21 47. The computer executable process steps stored on a computer readable medium of  
22 claim 35, wherein the predefined set of transformation algorithms  $L$  may be  
23 supplemented.  
24
- 25 48. The computer executable process steps stored on a computer readable medium of  
26 claim 35, wherein the selection of a number of conversion steps  $P$  may be a  
27 stochastic selection.  
28



- 1 49. The computer executable process steps stored on a computer readable medium of  
2 claim 35, wherein the selection of a number of conversion steps  $P$  may depend  
3 upon adherence to a mathematical criterion.  
4
- 5 50. The computer executable process steps stored on a computer readable medium of  
6 claim 35, wherein the selection of a number of conversion steps  $P$  may depend  
7 upon adherence to a logical criterion.  
8
- 9 51. The computer executable process steps stored on a computer readable medium of  
10 claim 35, wherein the selection of a number of conversion steps  $P$  may depend  
11 upon adherence to a file size criteria for encrypted final image  $G$ .  
12
- 13 52. The computer executable process steps stored on a computer readable medium of  
14 claim 35, wherein the alphabet of transformation  $AV$  is comprised of letters or  
15 quants, each letter or quant comprising a segment of structural data element  $S$ .  
16
- 17 53. The computer executable process steps stored on a computer readable medium of  
18 claim 52, wherein a number of bits in each letter or quant is stochastically  
19 selected.  
20
- 21 54. The computer executable process steps stored on a computer readable medium of  
22 claim 52, wherein a number of bits in each letter or quant may depend upon  
23 adherence to a mathematical criterion.  
24
- 25 55. The computer executable process steps stored on a computer readable medium of  
26 claim 52, wherein a number of bits in each letter or quant may depend upon  
27 adherence to a logical criterion.  
28

- 1 56. The computer executable process steps stored on a computer readable medium of  
2 claim 52, wherein a number of bits in each letter or quant may depend upon  
3 adherence to a file size criteria for encrypted final image  $G$ .  
4
- 5 57. The computer executable process steps stored on a computer readable medium of  
6 claim 35, wherein the information necessary to reverse the conversion process  
7 stored in tag data element  $T$  may comprise one or more of the following:  
8 an indicator of whether a current iterative step is the  $P^{th}$  iteration;  
9 an indicator of whether the selected external key  $K^x$  is to be used for all  $P$   
10 iterations;  
11 an indicator of the selected external key  $K^x$ ;  
12 an indicator of the selected transformation algorithm  $A$ ;  
13 a length of a first logical scale of position coding;  
14 the alphabet of transformation  $AV$ ; and  
15 other transformation algorithm  $A$  parameters.  
16
- 17 58. The computer executable process steps stored on a computer readable medium of  
18 claim 36, wherein the information necessary to reverse the conversion process  
19 stored in tag data element  $T$  may comprise one or more of the following:  
20 an indicator of whether a current iterative step is the  $P^{th}$  iteration;  
21 an indicator of whether the selected external key  $K^x$  is to be used for all  $P$   
22 iterations;  
23 an indicator of the selected external key  $K^x$ ;  
24 an indicator of the selected transformation algorithm  $A$ ;  
25 the alphabet of transformation  $AV$ ;  
26 a length of a first logical scale of position coding;  
27 other transformation algorithm  $A$  parameters;  
28 internal identifier  $K$  bit length and shift parameters; and  
29 an indicator of internal identifier  $K$  extraction.  
30

- 1 59. The computer executable process steps stored on a computer readable medium of  
2 claim 37, wherein the information necessary to reverse the conversion process  
3 stored in tag data element  $T$  may comprise one or more of the following:  
4 an indicator of whether a current iterative step is the  $P^{th}$  iteration;  
5 an indicator of whether the selected external key  $K^x$  is to be used for all  $P$   
6 iterations;  
7 an indicator of the selected external key  $K^x$ ;  
8 an indicator of the selected transformation algorithm  $A$ ;  
9 the alphabet of transformation  $AV$ ;  
10 a length of a first logical scale of position coding;  
11 other transformation algorithm  $A$  parameters;  
12 an indicator of the scrambling function selected;  
13 internal identifier  $K$  bit length and shift parameters; and  
14 an indicator of internal identifier  $K$  extraction.  
15
- 16 60. The computer executable process steps stored on a computer readable medium of  
17 claim 37, wherein the scrambling function is selected from a scrambling matrix  
18 comprised of a predefined set of scrambling functions.  
19
- 20 61. The computer executable process steps stored on a computer readable medium of  
21 claim 60, wherein the predefined set of scrambling functions is changed  
22 periodically.  
23
- 24 62. An apparatus for structurally converting a binary sequence into an encrypted final  
25 image  $G$ , comprising:  
26  
27 a memory element for storing computer executable process steps;  
28  
29 a processor for executing computer executable process steps;  
30

1 computer executable process steps comprising:

2

3 forming an image  $M$  of the binary sequence as a concatenation of a tag

4 data element  $T$  and structural data element  $S$ , tag data element  $T$

5 comprising information necessary to reverse a conversion process,

6 structural data element  $S$  comprising a sequence of logical scales of

7 position coding;

8

9 selecting a number of conversion steps  $P$  to be performed;

10

11 iteratively executing  $P$  times a conversion function comprised of the

12 following steps:

13 selecting a transformation algorithm  $A$  from a predefined set of

14 transformation algorithms  $L$ ;

15 selecting an alphabet of transformation  $AV$  based upon the

16 structural data element  $S$ ;

17 applying algorithm  $A$  and alphabet  $AV$  to structural data element  $S$

18 to form a plurality of logical scales of position coding;

19 forming a transformed structural data element  $S'$  comprised of a

20 sequence of the logical scales of position coding;

21 selecting an external key  $K^x$ ;

22 forming tag data element  $T$ ;

23 coding the tag data element  $T$  with external key  $K^x$  to obtain coded

24 tag data element  $T''$ ;

25 repeating the steps of the conversion function upon a converted

26 image  $M'$  comprised of a concatenation of the coded tag

27 data element  $T''$  and the transformed structural data element

28  $S'$ ;

1 and forming the encrypted final image  $G$  as a concatenation of the coded  
2 tag data element  $T''$  and the transformed structural data element  $S'$  created  
3 upon the  $P^{th}$  iteration of the conversion function.

4  
5 63. An apparatus for structurally converting a binary sequence into an encrypted final  
6 image  $G$ , comprising:

7  
8 a memory element for storing computer executable process steps;

9  
10 a processor for executing computer executable process steps;

11  
12 computer executable process steps comprising:

13 forming an image  $M$  of the binary sequence as a concatenation of a tag  
14 data element  $T$  and structural data element  $S$ , tag data element  $T$   
15 comprising information necessary to reverse the conversion process,  
16 structural data element  $S$  comprising a sequence of logical scales of  
17 position coding;  
18 selecting a number of conversion steps  $P$  to be performed;

19  
20 iteratively executing  $P$  times a conversion function comprised of the  
21 following steps:

22 selecting a transformation algorithm  $A$  from a predefined set of  
23 transformation algorithms  $L$ ;

24 selecting an alphabet of transformation  $AV$  based upon the  
25 structural data element  $S$ ;

26 applying algorithm  $A$  and alphabet  $AV$  to structural data element  $S$   
27 to form a plurality of logical scales of position coding;

28 forming a transformed structural data element  $S'$  comprised of a  
29 sequence of the logical scales of position coding;

30 stochastically selecting a bit length parameter and a shift parameter

1 which define an internal identifier  $K$  within transformed  
 2 structural data element  $S'$ ;  
 3 forming tag data element  $T$ ;  
 4 coding a portion of the tag data element  $T$  with internal identifier  $K$   
 5 to obtain a partially coded tag data element  $T'$ ;  
 6 selecting an external key  $K^x$ ;  
 7 coding the partially coded tag data element  $T'$  with external key  
 8  $K^x$  to obtain coded tag data element  $T''$ ;  
 9 stochastically determining whether to extract internal identifier  $K$   
 10 from transformed structural data element  $S'$ , and if  
 11 determined necessary, extracting the internal identifier  $K$   
 12 from transformed structural data element  $S'$  to obtain  
 13 structural data element  $S''$  and storing internal identifier  $K$   
 14 in a file of internal identifiers  $FID$ ;  
 15 performing the steps of the conversion function upon a converted  
 16 image  $M'$  comprised of a concatenation of the coded tag  
 17 data element  $T''$  and either transformed structural data  
 18 element  $S'$  if internal identifier  $K$  was not extracted, or  
 19 structural data element  $S''$  if internal identifier  $K$  was  
 20 extracted;  
 21  
 22 and forming the encrypted final image  $G$  as a concatenation of the coded tag  
 23 data element  $T''$  and either transformed structural data element  $S'$  if internal  
 24 identifier  $K$  was not extracted, or structural data element  $S''$  if internal  
 25 identifier  $K$  was extracted.

26  
 27 64. The apparatus of claim 63, wherein:

28  
 29 the processor is adapted to communicate on a network; and  
 30

1 the computer executable process steps further comprise:  
2  
3 structurally converting the file of internal identifiers  $FID$  to obtain a  
4 converted file of internal identifiers  $FID'$ , wherein a tag data  
5 element formed during the structural conversion of the file of  
6 internal identifiers  $FID$  is coded with an external key selected  
7 stochastically from a multitude of external keys in an external key  
8 file  $K_{EXT}$ ; and  
9 transmitting the encrypted final image  $G$  and structurally converted file of  
10 internal identifiers  $FID'$  to a subscriber or receiver.

11  
12 65. An apparatus for structurally converting a binary sequence into an encrypted final  
13 image  $G$ , comprising:

14  
15 a memory element for storing computer executable process steps;

16  
17 a processor for executing computer executable process steps;  
18 computer executable process steps comprising:

19  
20 forming an image  $M$  of the binary sequence as a concatenation of a tag  
21 data element  $T$  and structural data element  $S$ , tag data element  $T$   
22 comprising information necessary to reverse the conversion process,  
23 structural data element  $S$  comprising a sequence of logical scales of  
24 position coding;

25  
26 selecting a number of conversion steps  $P$  to be performed;

27  
28 iteratively executing  $P$  times a conversion function comprised of the  
29 following steps:

30



1 selecting a transformation algorithm  $A$  from a predefined set of  
2 transformation algorithms  $L$ ;  
3 selecting an alphabet of transformation  $AV$  based upon the  
4 structural data element  $S$ ;  
5 applying algorithm  $A$  and alphabet  $AV$  to structural data element  $S$   
6 to form a plurality of logical scales of position coding;  
7 forming a transformed structural data element  $S'$  comprised of a  
8 sequence of the logical scales of position coding;  
9 stochastically selecting a bit length parameter and a shift parameter  
10 which define an internal identifier  $K$  within transformed  
11 structural data element  $S'$ ;  
12 scrambling internal identifier  $K$  with a scrambling function to  
13 obtain a scrambled internal identifier  $K'$ ;  
14 forming tag data element  $T$ ;  
15 coding a portion of the tag data element  $T$  with scrambled internal  
16 identifier  $K'$  to obtain a partially coded tag data element  $T'$ ;  
17 selecting an external key  $K^x$ ;  
18 coding the partially coded tag data element  $T'$  with external key  
19  $K^x$  to obtain coded tag data element  $T''$ ;  
20 stochastically determining whether to extract internal identifier  $K$   
21 from transformed structural data element  $S'$ , and if  
22 determined necessary, extracting the internal identifier  $K$   
23 from transformed structural data element  $S'$  to obtain  
24 structural data element  $S''$  and storing scrambled internal  
25 identifier  $K'$  in a file of internal identifiers  $FID$ ;  
26 performing the steps of the conversion function upon a converted  
27 image  $M'$  comprised of a concatenation of the coded tag  
28 data element  $T''$  and either transformed structural data  
29 element  $S'$  if internal identifier  $K$  was not extracted, or



1                                structural data element  $S''$  if internal identifier  $K$  was  
2                                extracted;  
3                                and forming the encrypted final image  $G$  as a concatenation of the coded  
4                                tag data element  $T''$  and either transformed structural data element  $S'$  if  
5                                internal identifier  $K$  was not extracted, or structural data element  $S''$  if  
6                                internal identifier  $K$  was extracted.  
7  
8    66.    The apparatus of claim 65, wherein:  
9  
10           the processor is adapted to communicate on a network; and  
11  
12           the computer executable process steps further comprise:  
13                   structurally converting the file of internal identifiers  $FID$  to obtain a  
14                   converted file of internal identifiers  $FID'$ , wherein a tag data  
15                   element formed during the structural conversion of the file of  
16                   internal identifiers  $FID$  is coded with an external key selected  
17                   stochastically from a multitude of external keys in an external key  
18                   file  $K_{EXT}$ ; and  
19                   transmitting the encrypted final image  $G$  and structurally converted file of  
20                   internal identifiers  $FID'$  to a subscriber or receiver.  
21  
22    67.    The apparatus of claim 62, wherein the external key  $K^x$  is selected from a  
23           multitude of external keys in an external key file  $K_{EXT}$ .  
24  
25    68.    The apparatus of claim 62, wherein the selection of external key  $K^x$  is a stochastic  
26           selection.  
27  
28    69.    The apparatus of claim 62, wherein a same external key  $K^x$  is selected for use in  
29           all iterations.  
30

- 1     70.     The apparatus of claim 62, wherein a different external key  $K^x$  is selected upon  
2             each iteration.  
3
- 4     71.     The apparatus of claim 62, wherein the external key  $K^x$  is entered by a user during  
5             the conversion and reverse conversion process.  
6
- 7     72.     The apparatus of claim 67, wherein:  
8  
9             the processor is adapted to communicate on a network; and  
10  
11            the computer executable process steps further comprise:  
12                    structurally converting the external key file  $K_{EXT}$  to obtain a structurally  
13                    converted external key file; and  
14                    transmitting to a subscriber the structurally converted external key file and  
15                    an initial key  $K_{INIT}$  required to reverse the structural conversion of  
16                    the structurally converted external key file to obtain the external  
17                    key file  $K_{EXT}$ .  
18
- 19    73.     The apparatus of claim 62, wherein the selection of transformation algorithm  $A$   
20             may be a stochastic selection.  
21
- 22    74.     The apparatus of claim 62, wherein the selection of transformation algorithm  $A$   
23             may depend upon adherence to a mathematical criterion.  
24
- 25    75.     The apparatus of claim 62, wherein the selection of transformation algorithm  $A$   
26             may depend upon adherence to a logical criterion.  
27
- 28    76.     The apparatus of claim 62, wherein the selection of transformation algorithm  $A$   
29             may depend upon adherence to a file size criteria for encrypted final image  $G$ .  
30

- 1 77. The apparatus of claim 62, wherein the predefined set of transformation  
2 algorithms  $L$  may be supplemented.  
3
- 4 78. The apparatus of claim 62, wherein the selection of a number of conversion steps  
5  $P$  may be a stochastic selection.  
6
- 7 79. The apparatus of claim 62, wherein the selection of a number of conversion steps  
8  $P$  may depend upon adherence to a mathematical criterion.  
9
- 10 80. The apparatus of claim 62, wherein the selection of a number of conversion steps  
11  $P$  may depend upon adherence to a logical criterion.  
12
- 13 81. The apparatus of claim 62, wherein the selection of a number of conversion steps  
14  $P$  may depend upon adherence to a file size criteria for encrypted final image  $G$ .  
15
- 16 82. The apparatus of claim 62, wherein the alphabet of transformation  $AV$  is  
17 comprised of letters or quants, each letter or quant comprising a segment of  
18 structural data element  $S$ .  
19
- 20 83. The apparatus of claim 82, wherein a number of bits in each letter or quant is  
21 stochastically selected.  
22
- 23 84. The apparatus of claim 82, wherein a number of bits in each letter or quant may  
24 depend upon adherence to a mathematical criterion.  
25
- 26 85. The apparatus of claim 82, wherein a number of bits in each letter or quant may  
27 depend upon adherence to a logical criterion.  
28
- 29 86. The apparatus of claim 82, wherein a number of bits in each letter or quant may  
30 depend upon adherence to a file size criteria for encrypted final image  $G$ .

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30

87. The apparatus of claim 62, wherein the information necessary to reverse the conversion process stored in tag data element  $T$  may comprise one or more of the following:

- an indicator of whether a current iterative step is the  $P^{th}$  iteration;
- an indicator of whether the selected external key  $K^x$  is to be used for all  $P$  iterations;
- an indicator of the selected external key  $K^x$ ;
- an indicator of the selected transformation algorithm  $A$ ;
- a length of a first logical scale of position coding;
- the alphabet of transformation  $AV$ ; and
- other transformation algorithm  $A$  parameters.

88. The apparatus of claim 63, wherein the information necessary to reverse the conversion process stored in tag data element  $T$  may comprise one or more of the following:

- an indicator of whether a current iterative step is the  $P^{th}$  iteration;
- an indicator of whether the selected external key  $K^x$  is to be used for all  $P$  iterations;
- an indicator of the selected external key  $K^x$ ;
- an indicator of the selected transformation algorithm  $A$ ;
- the alphabet of transformation  $AV$ ;
- a length of a first logical scale of position coding;
- other transformation algorithm  $A$  parameters;
- the bit internal identifier  $K$  length and shift parameters; and
- an indicator of internal identifier  $K$  extraction.

89. The apparatus of claim 63, wherein the information necessary to reverse the conversion process stored in tag data element  $T$  may comprise one or more of the following:

1 an indicator of whether a current iterative step is the  $P^{th}$  iteration;  
 2 an indicator of whether the selected external key  $K^x$  is to be used for all  $P$   
 3 iterations;  
 4 an indicator of the selected external key  $K^x$ ;  
 5 an indicator of the selected transformation algorithm  $A$ ;  
 6 the alphabet of transformation  $AV$ ;  
 7 a length of a first logical scale of position coding;  
 8 other transformation algorithm  $A$  parameters;  
 9 an indicator of the scrambling function selected;  
 10 the bit internal identifier  $K$  length and shift parameters; and  
 11 an indicator of internal identifier  $K$  extraction.

12  
 13 90. The apparatus of claim 65, wherein the scrambling function is selected from a  
 14 scrambling matrix comprised of a predefined set of scrambling functions.

15  
 16 91. The apparatus of claim 90, wherein the predefined set of scrambling functions is  
 17 changed periodically.

18  
 19  
 20